Vanskelige beregninger av enkle utsagn

Herman Ruge Jervell

OMS–seminar
11.04.2008

Abstract

Det er ikke så veldig interessant med enkle utsagn som kan beregnes enkelt. Heller ikke kompliserte utsagn som må beregnes komplisert. Her ser vi på enkle utsagn som trenger kompliserte og vanskelige beregninger. Jeg vil gi eksempler på dette og vise tilknytning til problemer i kompleksitet og ufullstendighet.

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.
Dette er den måten vi anskueliggjør store tall.

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.
Dette er den måten vi anskueliggjør store tall.

- $10 =$ to hender

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.
Dette er den måten vi anskueliggjør store tall.

- $10 =$ to hender
- $11 =$ en til overs

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.
Dette er den måten vi anskueliggjør store tall.

- $10 =$ to hender
- $11 =$ en til overs
- $12 =$ to til overs

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.
Dette er den måten vi anskueliggjør store tall.

- $10 =$ to hender
- $11 =$ en til overs
- $12 =$ to til overs
- $100 =$ to hender av to hender

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.
Dette er den måten vi anskueliggjør store tall.

- $10 =$ to hender
- $11 =$ en til overs
- $12 =$ to til overs
- $100 =$ to hender av to hender
- $1000 =$ stort hundre

Arkimedes

Å telle med store tall

I en liten artikkel på 3 sider viste Arkimedes hvordan en kunne telle antall sandkorn i universet.
Han talte opp til 10^{80}. Først telle opp til 10. Så ta 10-erskritt for å telle opp til 100. Og så videre.
Dette er den måten vi anskueliggjør store tall.

- $10 =$ to hender
- $11 =$ en til overs
- $12 =$ to til overs
- $100 =$ to hender av to hender
- $1000 =$ stort hundre

De naturlige tall

- Predikat $\mathcal{N}$ – skriver $x : \mathcal{N}$ eller $\mathcal{N}x$

De naturlige tall

- Predikat $\mathcal{N}$ – skriver $x : \mathcal{N}$ eller $\mathcal{N}x$
- Start $0 : \mathcal{N}$

De naturlige tall

- Predikat $\mathcal{N}$ – skriver $x : \mathcal{N}$ eller $\mathcal{N}x$
- Start $0 : \mathcal{N}$
- Unær konstruktor $s : \mathcal{N} \to \mathcal{N}$

De naturlige tall

- Predikat $\mathcal{N}$ – skriver $x : \mathcal{N}$ eller $\mathcal{N}x$
- Start $0 : \mathcal{N}$
- Unær konstruktor $s : \mathcal{N} \rightarrow \mathcal{N}$

De naturlige tall

- Predikat $\mathcal{N}$ – skriver $x : \mathcal{N}$ eller $\mathcal{N}x$
- Start $0 : \mathcal{N}$
- Unær konstruktor $s : \mathcal{N} \rightarrow \mathcal{N}$

Dette er de unære tall. I tillegg har vi en notasjon $exy = 2^x + y$ gitt ved

De naturlige tall

- Predikat $\mathcal{N}$ – skriver $x : \mathcal{N}$ eller $\mathcal{N}x$
- Start $0 : \mathcal{N}$
- Unær konstruktor $s : \mathcal{N} \to \mathcal{N}$

Dette er de unære tall. I tillegg har vi en notasjon $exy = 2^x + y$ gitt ved

De naturlige tall

- Predikat $\mathcal{N}$ – skriver $x : \mathcal{N}$ eller $\mathcal{N}x$
- Start $0 : \mathcal{N}$
- Unær konstruktor $s : \mathcal{N} \rightarrow \mathcal{N}$

Dette er de unære tall. I tillegg har vi en notasjon $exy = 2^x + y$ gitt ved

$$
\begin{aligned}
e0y &= sy \; (\; = 2^0 + y) \\
esxy &= exexy \; (\; = 2^x + 2^x + y)
\end{aligned}
$$

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

- $1 = s0 = e00$

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

- $1 = s0 = e00$
- $2 = ss0 = e10 = ee000$

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

- $1 = s0 = e00$
- $2 = ss0 = e10 = ee000$
- $3 = e11 = ee00e00$

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

- $1 = s0 = e00$
- $2 = ss0 = e10 = ee000$
- $3 = e11 = ee00e00$
- $4 = e20 = eee0000$

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

- $1 = s0 = e00$
- $2 = ss0 = e10 = ee000$
- $3 = e11 = ee00e00$
- $4 = e20 = eee0000$
- $16 = e40 = eeee00000 = ssssssssssssssss0$

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

- $1 = s0 = e00$
- $2 = ss0 = e10 = ee000$
- $3 = e11 = ee00e00$
- $4 = e20 = eee0000$
- $16 = e40 = eeee00000 = ssssssssssssssss0$
- $256 = 2^{16} = eeeee000000$

Notasjonssystem

Med $\mathcal{N}$ og 0 og sx har vi definert den aller enkleste uendelige datastrukturen.

Med exy får vi definert et flott notasjonssystem. Med det får vi lagd korte beskrivelser av store tall

- $1 = s0 = e00$
- $2 = ss0 = e10 = ee000$
- $3 = e11 = ee00e00$
- $4 = e20 = eee0000$
- $16 = e40 = eeee00000 = ssssssssssssssss0$
- $256 = 2^{16} = eeeee000000$
- $2^{256} = eeeeee0000000$ atomer i universet

Problem

Hvorfor er $eeeeee0000000$ et tall?

Problem

Hvorfor er $eeeeee0000000$ et tall?

Hvordan kan vi vise det?

Problem

Hvorfor er *eeeeee*0000000 et tall?

Hvordan kan vi vise det?

Kan vi vise det mekanisk?

Aksiomer

Aksiomer

DATASTRUKTUR

Aksiomer

DATASTRUKTUR

$$0 : \mathcal{N} \tag{1}$$

$$x : \mathcal{N} \rightarrow sx : \mathcal{N} \tag{2}$$

Aksiomer

DATASTRUKTUR

$$0 : \mathcal{N} \tag{1}$$
$$x : \mathcal{N} \rightarrow sx : \mathcal{N} \tag{2}$$

NOTASJON

Aksiomer

DATASTRUKTUR

$$0 : \mathcal{N} \tag{1}$$
$$x : \mathcal{N} \to sx : \mathcal{N} \tag{2}$$

NOTASJON

$$e0y = sy \tag{3}$$
$$esxy = exexy \tag{4}$$

Aksiomer

DATASTRUKTUR

$$0 : \mathcal{N} \tag{1}$$
$$x : \mathcal{N} \rightarrow sx : \mathcal{N} \tag{2}$$

NOTASJON

$$e0y = sy \tag{3}$$
$$esxy = exexy \tag{4}$$

Kan vi vise

$$\mathrm{DATASTRUKTUR} \wedge \mathrm{NOTASJON} \rightarrow eeeeee0000000 : \mathcal{N}$$

Dårlig strategi

- Bruk DATASTRUKTUR til å telle opp til tallet

Dårlig strategi

- Bruk DATASTRUKTUR til å telle opp til tallet
- Deretter NOTASJON til å forenkle svaret

Dårlig strategi

- Bruk DATASTRUKTUR til å telle opp til tallet
- Deretter NOTASJON til å forenkle svaret

Trenger da bevis som er større enn universet.

Dårlig strategi

- Bruk DATASTRUKTUR til å telle opp til tallet
- Deretter NOTASJON til å forenkle svaret

Trenger da bevis som er større enn universet.

Mekaniske bevisere bruker denne dårlige strategien.

Hjelpepredikat

$\mathcal{N}_i$ definert ved

$$\mathcal{N}_0 = \mathcal{N} \tag{5}$$
$$x : \mathcal{N}_{k+1} = \forall y : \mathcal{N}_k . exy : \mathcal{N}_k \tag{6}$$

Hva betyr hjelpepredikatet?

$$0 : \mathcal{N}_1 \Leftrightarrow \forall x : \mathcal{N}.x + 2^0 : \mathcal{N}$$
$$1 : \mathcal{N}_1 \Leftrightarrow \forall x : \mathcal{N}.x + 2^1 : \mathcal{N}$$
$$2 : \mathcal{N}_1 \Leftrightarrow \forall x : \mathcal{N}.x + 2^2 : \mathcal{N}$$
$$3 : \mathcal{N}_1 \Leftrightarrow \forall x : \mathcal{N}.x + 2^3 : \mathcal{N}$$
$$\ldots$$
$$0 : \mathcal{N}_2 \Leftrightarrow \forall x : \mathcal{N}_1.x + 2^0 : \mathcal{N}_1$$
$$1 : \mathcal{N}_2 \Leftrightarrow \forall x : \mathcal{N}_1.x + 2^1 : \mathcal{N}_1$$
$$2 : \mathcal{N}_2 \Leftrightarrow \forall x : \mathcal{N}_1.x + 2^2 : \mathcal{N}_1$$
$$3 : \mathcal{N}_2 \Leftrightarrow \forall x : \mathcal{N}_1.x + 2^3 : \mathcal{N}_1$$
$$\ldots$$

Hva betyr hjelpepredikatet?

$$
\begin{aligned}
0 : \mathcal{N}_1 &\Leftrightarrow \forall x : \mathcal{N}.x + 2^0 : \mathcal{N} \\
1 : \mathcal{N}_1 &\Leftrightarrow \forall x : \mathcal{N}.x + 2^1 : \mathcal{N} \\
2 : \mathcal{N}_1 &\Leftrightarrow \forall x : \mathcal{N}.x + 2^2 : \mathcal{N} \\
3 : \mathcal{N}_1 &\Leftrightarrow \forall x : \mathcal{N}.x + 2^3 : \mathcal{N} \\
&\dots \\
0 : \mathcal{N}_2 &\Leftrightarrow \forall x : \mathcal{N}_1.x + 2^0 : \mathcal{N}_1 \\
1 : \mathcal{N}_2 &\Leftrightarrow \forall x : \mathcal{N}_1.x + 2^1 : \mathcal{N}_1 \\
2 : \mathcal{N}_2 &\Leftrightarrow \forall x : \mathcal{N}_1.x + 2^2 : \mathcal{N}_1 \\
3 : \mathcal{N}_2 &\Leftrightarrow \forall x : \mathcal{N}_1.x + 2^3 : \mathcal{N}_1 \\
&\dots
\end{aligned}
$$

Hver linje — 2 ganger linja over

Hjelpesetning

Lemma

$$\mathrm{DATASTRUKTUR} \wedge \mathrm{NOTASJON} \rightarrow 0 : \mathcal{N}_i$$

Hjelpesetning

Lemma

$$\text{DATASTRUKTUR} \wedge \text{NOTASJON} \rightarrow 0 : \mathcal{N}_i$$

Opplagt for $i = 0$ og $i = 1$. Vis for $i + 2$.

Hjelpesetning

Lemma

$$\text{DATASTRUKTUR} \wedge \text{NOTASJON} \rightarrow 0 : \mathcal{N}_i$$

Opplagt for $i = 0$ og $i = 1$. Vis for $i + 2$.
Skal vise: $\forall x : \mathcal{N}_{i+1}.sx : \mathcal{N}_{i+1}$

Hjelpesetning

Lemma

$$\text{DATASTRUKTUR} \wedge \text{NOTASJON} \rightarrow 0 : \mathcal{N}_i$$

Opplagt for $i = 0$ og $i = 1$. Vis for $i + 2$.
Skal vise: $\forall x : \mathcal{N}_{i+1}.sx : \mathcal{N}_{i+1}$
Anta: $a : \mathcal{N}_{i+1}$ dvs $\forall u : \mathcal{N}_i.eau : \mathcal{N}_i$

Hjelpesetning

Lemma

$$\text{DATASTRUKTUR} \wedge \text{NOTASJON} \rightarrow 0 : \mathcal{N}_i$$

Opplagt for $i = 0$ og $i = 1$. Vis for $i + 2$.
Skal vise: $\forall x : \mathcal{N}_{i+1}.sx : \mathcal{N}_{i+1}$
Anta: $a : \mathcal{N}_{i+1}$ dvs $\forall u : \mathcal{N}_i.eau : \mathcal{N}_i$
Anta: $b : \mathcal{N}_i$.

Hjelpesetning

Lemma

$$\text{DATASTRUKTUR} \wedge \text{NOTASJON} \rightarrow 0 : \mathcal{N}_i$$

Opplagt for $i = 0$ og $i = 1$. Vis for $i + 2$.
Skal vise: $\forall x : \mathcal{N}_{i+1}.sx : \mathcal{N}_{i+1}$
Anta: $a : \mathcal{N}_{i+1}$ dvs $\forall u : \mathcal{N}_i.eau : \mathcal{N}_i$
Anta: $b : \mathcal{N}_i$.
Men da $eab : \mathcal{N}_i$ og $esab = eaeab : \mathcal{N}_i$ ved å bruke $a : \mathcal{N}_{i+1}$ to ganger.

Hjelpesetning

Lemma

$$\mathrm{DATASTRUKTUR} \wedge \mathrm{NOTASJON} \rightarrow 0 : \mathcal{N}_i$$

Opplagt for $i = 0$ og $i = 1$. Vis for $i + 2$.
Skal vise: $\forall x : \mathcal{N}_{i+1}.sx : \mathcal{N}_{i+1}$
Anta: $a : \mathcal{N}_{i+1}$ dvs $\forall u : \mathcal{N}_i.eau : \mathcal{N}_i$
Anta: $b : \mathcal{N}_i$.
Men da $eab : \mathcal{N}_i$ og $esab = eaeab : \mathcal{N}_i$ ved å bruke $a : \mathcal{N}_{i+1}$ to ganger.
Dermed $sa : \mathcal{N}_{i+1}$ for enhver $a : \mathcal{N}_{i+1}$ og $0 : \mathcal{N}_{i+2}$

God strategi

God strategi

- $0 : \mathcal{N}_6$ og $0 : \mathcal{N}_5$ gir $e00 : \mathcal{N}_5$

God strategi

- $0 : \mathcal{N}_6$ og $0 : \mathcal{N}_5$ gir $e00 : \mathcal{N}_5$
- som sammen med $0 : \mathcal{N}_4$ gir $ee000 : \mathcal{N}_4$

God strategi

- $0 : \mathcal{N}_6$ og $0 : \mathcal{N}_5$ gir $e00 : \mathcal{N}_5$
- som sammen med $0 : \mathcal{N}_4$ gir $ee000 : \mathcal{N}_4$
- som sammen med $0 : \mathcal{N}_3$ gir $eee0000 : \mathcal{N}_3$

God strategi

- $0 : \mathcal{N}_6$ og $0 : \mathcal{N}_5$ gir $e00 : \mathcal{N}_5$
- som sammen med $0 : \mathcal{N}_4$ gir $ee000 : \mathcal{N}_4$
- som sammen med $0 : \mathcal{N}_3$ gir $eee0000 : \mathcal{N}_3$
- som sammen med $0 : \mathcal{N}_2$ gir $eeee00000 : \mathcal{N}_2$

God strategi

- $0 : \mathcal{N}_6$ og $0 : \mathcal{N}_5$ gir $e00 : \mathcal{N}_5$
- som sammen med $0 : \mathcal{N}_4$ gir $ee000 : \mathcal{N}_4$
- som sammen med $0 : \mathcal{N}_3$ gir $eee0000 : \mathcal{N}_3$
- som sammen med $0 : \mathcal{N}_2$ gir $eeee00000 : \mathcal{N}_2$
- som sammen med $0 : \mathcal{N}_1$ gir $eeeee000000 : \mathcal{N}_1$

God strategi

- $0 : \mathcal{N}_6$ og $0 : \mathcal{N}_5$ gir $e00 : \mathcal{N}_5$
- som sammen med $0 : \mathcal{N}_4$ gir $ee000 : \mathcal{N}_4$
- som sammen med $0 : \mathcal{N}_3$ gir $eee0000 : \mathcal{N}_3$
- som sammen med $0 : \mathcal{N}_2$ gir $eeee00000 : \mathcal{N}_2$
- som sammen med $0 : \mathcal{N}_1$ gir $eeeee000000 : \mathcal{N}_1$
- som sammen med $0 : \mathcal{N}_0$ gir $eeeeee0000000 : \mathcal{N}_0$

God strategi

- $0 : \mathcal{N}_6$ og $0 : \mathcal{N}_5$ gir $e00 : \mathcal{N}_5$
- som sammen med $0 : \mathcal{N}_4$ gir $ee000 : \mathcal{N}_4$
- som sammen med $0 : \mathcal{N}_3$ gir $eee0000 : \mathcal{N}_3$
- som sammen med $0 : \mathcal{N}_2$ gir $eeee00000 : \mathcal{N}_2$
- som sammen med $0 : \mathcal{N}_1$ gir $eeeee000000 : \mathcal{N}_1$
- som sammen med $0 : \mathcal{N}_0$ gir $eeeeee0000000 : \mathcal{N}_0$
- som skulle vises

Diskusjon

Diskusjon

- Hvor godt er eksemplet?

Diskusjon

- Hvor godt er eksemplet?
- Bruk av likhet

Diskusjon

- Hvor godt er eksemplet?
- Bruk av likhet
- Andre notasjoner

Diskusjon

- Hvor godt er eksemplet?
- Bruk av likhet
- Andre notasjoner
- Finne hjelpesetninger

Diskusjon

- Hvor godt er eksemplet?
- Bruk av likhet
- Andre notasjoner
- Finne hjelpesetninger
- Hva er viktig i hjelpesetningene

Diskusjon

- Hvor godt er eksemplet?
- Bruk av likhet
- Andre notasjoner
- Finne hjelpesetninger
- Hva er viktig i hjelpesetningene
- Hjelpesetninger til interaktive bevis

Diskusjon

- Hvor godt er eksemplet?
- Bruk av likhet
- Andre notasjoner
- Finne hjelpesetninger
- Hva er viktig i hjelpesetningene
- Hjelpesetninger til interaktive bevis
- Andre datastrukturer

Endelige trær

Vi kan representere naturlige tall

Endelige trær

Vi kan representere naturlige tall

$$0 = \ .$$

Endelige trær

Vi kan representere naturlige tall

$$0 = \;. \qquad\qquad 1 = \begin{array}{c} . \\ | \\ . \end{array}$$

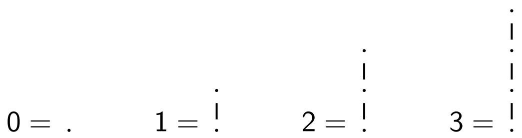
Endelige trær

Vi kan representere naturlige tall

$$0 = \;\cdot \qquad\qquad 1 = \begin{array}{c} \cdot \\ | \\ \cdot \end{array} \qquad\qquad 2 = \begin{array}{c} \cdot \\ | \\ \cdot \\ | \\ \cdot \end{array}$$

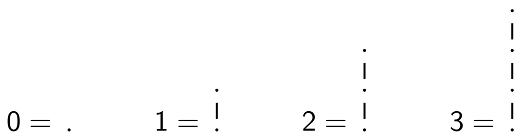
Endelige trær

Vi kan representere naturlige tall



Endelige trær

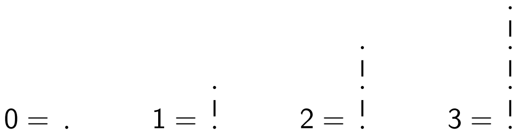
Vi kan representere naturlige tall



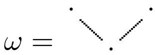
Og noen uendelige ordinaltall:

Endelige trær

Vi kan representere naturlige tall

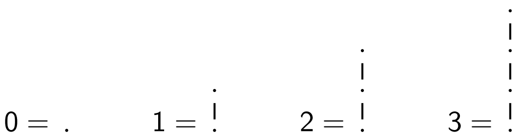


Og noen uendelige ordinaltall:

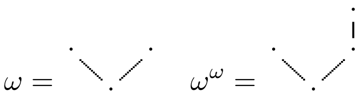


Endelige trær

Vi kan representere naturlige tall

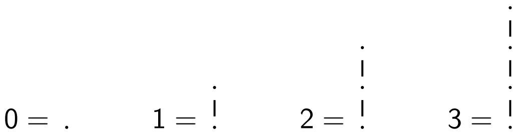


Og noen uendelige ordinaltall:

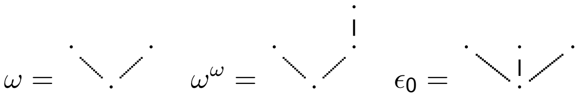


Endelige trær

Vi kan representere naturlige tall

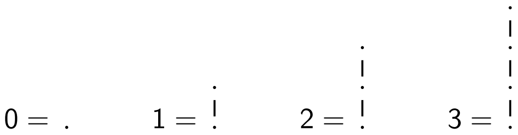


Og noen uendelige ordinaltall:

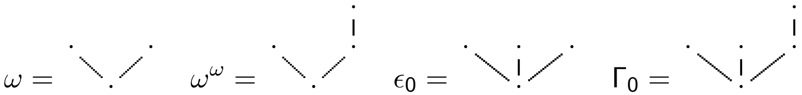


Endelige trær

Vi kan representere naturlige tall



Og noen uendelige ordinaltall:



Trærne har rot og greinene er ordnet.

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\Leftrightarrow\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \quad \Leftrightarrow \quad \mathbf{A} \leq \langle \mathbf{B} \rangle \ \vee \ (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\Leftrightarrow\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\Leftrightarrow\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\Leftrightarrow\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle \mathbf{A} \rangle < \mathbf{B}$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\Leftrightarrow\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle \mathbf{A} \rangle < \mathbf{B}$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\Leftrightarrow\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle \mathbf{A} \rangle < \mathbf{B}$ for alle umiddelbare deltrær av $\mathbf{A}$ er $< \mathbf{B}$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\; \Leftrightarrow \;\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle \mathbf{A} \rangle < \mathbf{B}$ for alle umiddelbare deltrær av $\mathbf{A}$ er $< \mathbf{B}$
- $\langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\;\Leftrightarrow\;\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \;\wedge\; \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle \mathbf{A} \rangle < \mathbf{B}$ for alle umiddelbare deltrær av $\mathbf{A}$ er $< \mathbf{B}$
- $\langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle$

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\Leftrightarrow\; \mathbf{A} \leq \langle\mathbf{B}\rangle \;\vee\; (\langle\mathbf{A}\rangle < \mathbf{B} \wedge \langle\mathbf{A}\rangle < \langle\mathbf{B}\rangle)}$$

- $\langle\mathbf{A}\rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle\mathbf{B}\rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle\mathbf{A}\rangle < \mathbf{B}$ for alle umiddelbare deltrær av $\mathbf{A}$ er $< \mathbf{B}$
- $\langle\mathbf{A}\rangle < \langle\mathbf{B}\rangle$ leksikografisk ordnet

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\; \Leftrightarrow \;\; \mathbf{A} \le \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \le \langle \mathbf{B} \rangle$ $\mathbf{A} \le$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle \mathbf{A} \rangle < \mathbf{B}$ for alle umiddelbare deltrær av $\mathbf{A}$ er $< \mathbf{B}$
- $\langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle$ leksikografisk ordnet
  - Lengden av følgene

Ordne trærne

$$\boxed{\mathbf{A} < \mathbf{B} \;\; \Leftrightarrow \;\; \mathbf{A} \leq \langle \mathbf{B} \rangle \;\vee\; (\langle \mathbf{A} \rangle < \mathbf{B} \wedge \langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle)}$$

- $\langle \mathbf{A} \rangle$ — følgen av umiddelbare deltrær
- $\mathbf{A} \leq \langle \mathbf{B} \rangle$ $\mathbf{A} \leq$ for et umiddelbart deltre av $\mathbf{B}$
- $\langle \mathbf{A} \rangle < \mathbf{B}$ for alle umiddelbare deltrær av $\mathbf{A}$ er $< \mathbf{B}$
- $\langle \mathbf{A} \rangle < \langle \mathbf{B} \rangle$ leksikografisk ordnet
  - Lengden av følgene
  - Ytterste høyre element der de er forskjellige

Beskrive prosesser

Beskrive prosesser

$\omega =$ 

Beskrive prosesser

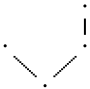
$\omega =$ 

Det minste tre større enn alle naturlige tall

Beskrive prosesser

$\omega =$ 

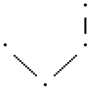
Det minste tre større enn alle naturlige tall

$\omega^\omega =$ 

Beskrive prosesser

$\omega =$ 

Det minste tre større enn alle naturlige tall

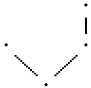
$\omega^\omega =$ 

Til å beskrive programmer med for-løkker

Beskrive prosesser

$\omega =$ 

Det minste tre større enn alle naturlige tall

$\omega^\omega =$ 

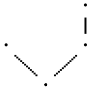
Til å beskrive programmer med for-løkker

$\epsilon_0 =$ 

Beskrive prosesser

$\omega =$ 

Det minste tre større enn alle naturlige tall

$\omega^\omega =$ 

Til å beskrive programmer med for-løkker

$\epsilon_0 =$ 

Gödels ufullstendighets teorem for elementær aritmetikk

FIN